

基于 3DGS 的大场景实例分割探索

摘要

摘要：真实场景渲染是图形学的研究热点，相关研究被广泛应用于智慧城市，三维场景生成，智能驾驶等场景。在过去，场景渲染主要依赖多视角合成技术实现多视角一致性与真实渲染，近年来基于隐式表达的辐射场方法在真实场景渲染上表现出了优秀的性能与渲染表现。本工作在大场景实时渲染技术基础上，通过引入基础大模型，尝试在预训练大场景下进行实例分割渲染，并围绕大大场景渲染模型构建可视化程序用于用户高效交互与对比分析。

关键词：渲染；点云模型；光栅化渲染；机器学习

1 实验课题介绍

1.1 课题背景

1.1.1 传统渲染管线

传统渲染管线设计将模型渲染流程进行解耦，实现模型数据的并行处理与定制化工作。一般来说，渲染流程被划分为顶点处理，几何处理，像素处理。实际渲染处理过程中，模型一般以顶点坐标及顶点连接关系信息输入渲染器中，顶点处理器首先对原始模型顶点位置进行批处理，包括顶点集合划分，曲面嵌入，曲面细分，从而对模型曲面区域进行划分与细分用于曲面优化处理；几何处理器对原始模型信息进行变换，将其转换为实际成像平面的坐标，几何处理器还会对模型进行投影变换，使用基于距离等视锥提取技术对模型原始顶点进行预选择，实现对原始模型的重投影。光栅化渲染器缓存了一个像素网格占用情况，其中每个像素都有对应的透明度，颜色值，通过将三维体素数据投影到屏幕空间并进行进一步的处理，从而在保证性能优化的情况下对场景信息实现更逼真的渲染。

1.1.2 离线渲染与实时渲染

传统渲染可以划分为离线渲染与实时渲染两种渲染形式，它们的渲染目标及实现技术路线均有较大差异。离线渲染通过在渲染过程中使用复杂的非线性拟合方法 [15] 及动力系统，从而实现精细的粒子交互 [3]，复杂光照模型及高质量高真实感的图像。尽管离线渲染可以实现高质量渲染结果，但其使用的路径追踪、全局光照、复杂方程拟合迭代求解等技术对处理设备有极高要求，且渲染时间极长，无法实现交互需求。实时渲染则聚焦于可交互与实时显示，这种技术常被应用在游戏、虚拟现实等强交互的应用场景，因此其必须确保实时 (+30fps) 的

渲染速度及流畅的体验。基于以上需求，实时渲染对传统的渲染管线进行两方面的优化，其一是对渲染方案的优化，通过对物理模型简化，在合理性错误容忍的基础上对模型精度及光照模型进行简化与离散化，实现大幅下降渲染的时间。其二是对渲染流程的模块化，解耦与并行化 [4]。通过对通用计算模块的高效利用，实现成像平面的并行渲染计算及后处理，通过将场景信息进行预选择与缓存，在渲染流程解耦的基础上进一步提升了渲染速度。

1.1.3 显式表示与隐式表示

显式表示和隐式表示是计算机图形学、几何建模以及其他数学领域中常用的两种表示方法，它们在数据存储方式、计算处理技术和应用场景上有显著区别。显式表示通过对模型给出准确的几何描述实现对模型的表示与保存。显式表示主要特点有三，其一是直接描述，显式表示通过一些显式的几何数据来描述物体的形状和结构，因此对模型的遍历与访问便捷快速。其二是操作处理便捷，显式表示方法使模型可以快速直接进行几何变换（如平移、旋转、缩放）或者进行细分、简化等操作。其三是网格表示的快速转换，光栅化渲染器普遍采用网格缓存，需要将模型进行处理，显式表示使模型可以很好地转换成体素占用的情况，辅助光栅化渲染。隐式表示通过隐式方程或者约束来描述物体，其不直接给出物体的形状，而是通过某种数学方程来间接地表示物体的存在。隐式表示可以自然地定义物体内外空间，且这种表示具有分辨率非依赖性。对于拓扑变化和复杂的形状，隐式表示可以更灵活地处理这种结构。隐式表示具有的连续性与自然的空间约束性使其成为深度学习拟合的优质对象，通过使用深层网络对齐进行学习，研究人员可以实现对模型高维结构的预测 [7]。

2 相关工作

2.1 传统场景重建与渲染

最早的虚拟视图合成方法基于光场，最初是通过密集采样来实现的 [6, 12]。运动恢复结构 (SfM) 的出现 [16] 开启了一个全新的领域，如图一所示，在这个领域中，一组照片可以用来合成新的视角。SfM 在相机标定过程中估计一个稀疏的点云，这些点云最初用于简单的三维空间可视化。随后，多视图立体 (MVS) 技术发展出了令人印象深刻的完整三维重建算法 [5]，并推动了多个视角合成算法的发展。这些方法通过将输入图像重新投影并融合到新的视角相机中，利用几何信息来引导这一重新投影过程。这些方法在许多情况下取得了优秀的成果，但仍存在许多未重建区域，或者在“过度重建”时产生不存在的几何体（即 MVS 生成了虚假的几何结构）。近年来的神经渲染算法 [17] 大大减少了这种伪影，并避免了在 GPU 上存储所有输入图像的巨大开销，在大多数方面超越了这些传统方法。

2.2 神经渲染与辐射场

深度学习技术早期便被用于新视角合成。卷积神经网络 (CNN) 被用来估计融合权重 [8–10]，或用于纹理空间的解决方案。大多数这些方法的主要缺点是依赖于基于多视图立体 (MVS) 的几何信息；此外，CNN 用于最终渲染时，常常会导致时间上的闪烁。用于新视角合成的体积表示方法由 Soft3D [14] 开创；随后，结合深度学习技术和体积光线行进 (volumetric ray-marching) 的方法相继被提出 [11]，它们基于连续可微的密度场来表示几何体。使用体积光线

行进渲染具有显著的计算成本，因为需要大量样本来查询体积。神经辐射场（NeRF）[13] 引入了重要性采样和位置编码来提升质量，但其使用的大型多层感知器（MLP）影响了渲染速度。NeRF 的成功引发了大量后续方法的涌现，这些方法通过引入正则化策略来提高质量和速度；目前，新视角合成在图像质量方面的最新成果是 Mip-NeRF360 [1,2]。虽然渲染质量非常优秀，但训练和渲染时间依然非常高。最新的方法主要通过利用三种设计选择来加速训练和/或渲染：使用空间数据结构来存储（神经）特征，随后在体积光线行进中进行插值，使用不同的编码方式，以及调整 MLP 的容量。这些方法包括空间离散化的不同变种，及哈希表。其中最著名的方法包括 InstantNGP，该方法使用哈希网格和占用网格加速计算，并使用更小的 MLP 来表示密度和外观；以及 Plenoxels [Fridovich-Keil and Yu et al. 2022]，该方法使用稀疏体素网格来插值连续的密度场，并能够完全舍弃神经网络。两者都依赖于球面谐波：前者直接表示方向效应，后者则用于对输入进行编码并传递给颜色网络。虽然这两种方法都提供了优秀的结果，但它们在有效表示空旷空间方面仍然存在挑战，这在一定程度上取决于场景或捕捉类型。此外，图像质量在很大程度上受到加速使用的结构化网格选择的限制，而渲染速度则受到每次光线行进步骤需要查询大量样本的影响。

2.3 基于高斯喷溅实现实时渲染

高斯喷溅技术对传统点云进行扩展。通过对输入记得静态场景图像进行特征提取，得到稀疏的初始点云。进一步高斯喷溅构建了一组高斯函数，如图二所示通过位置，均方差矩阵和不透明度定义每个高斯椭球的颜色及场景表示，进而进行优化机制设计。基于以上的表示设计，高斯喷溅技术通过将场景的表示元素进行各向异性改在实现了紧凑的三维场景表示结构。进一步，高斯喷溅设计了一系列优化不走，通过对高斯函数的位置信息、协方差矩阵、和 SH 系数进行优化，并再此过程中自适应地控制高斯密度。

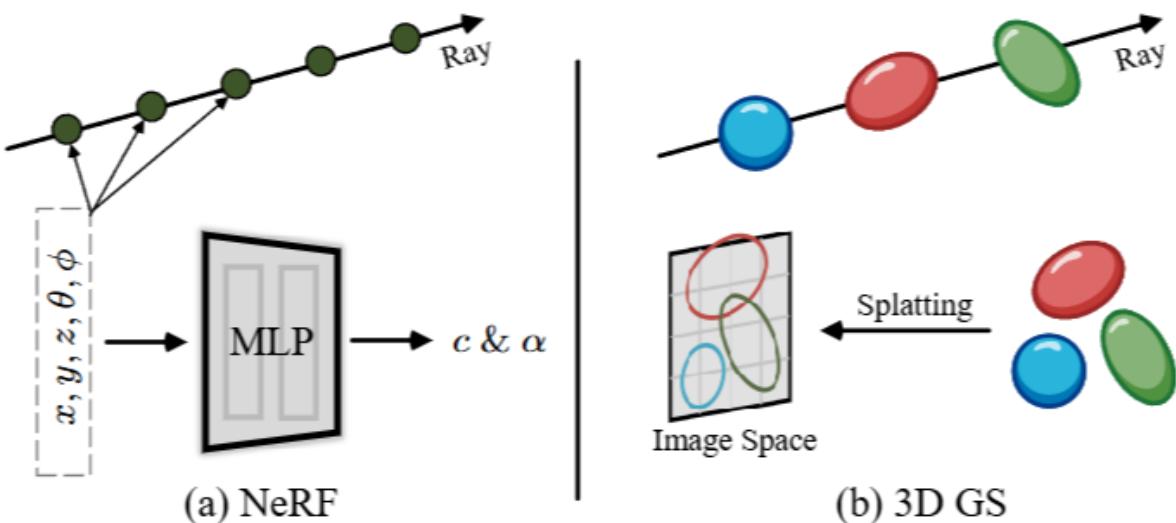


图 1. 高斯喷溅技术与神经辐射场技术对比

对于场景渲染，高斯喷溅在传统渲染管线的基础上，设计了基于成像平面划分的光栅化器，通过采用体积渲染技术，对各项异性的高斯椭圆进行混合，并通过快速排序确保正确的可见性顺序。

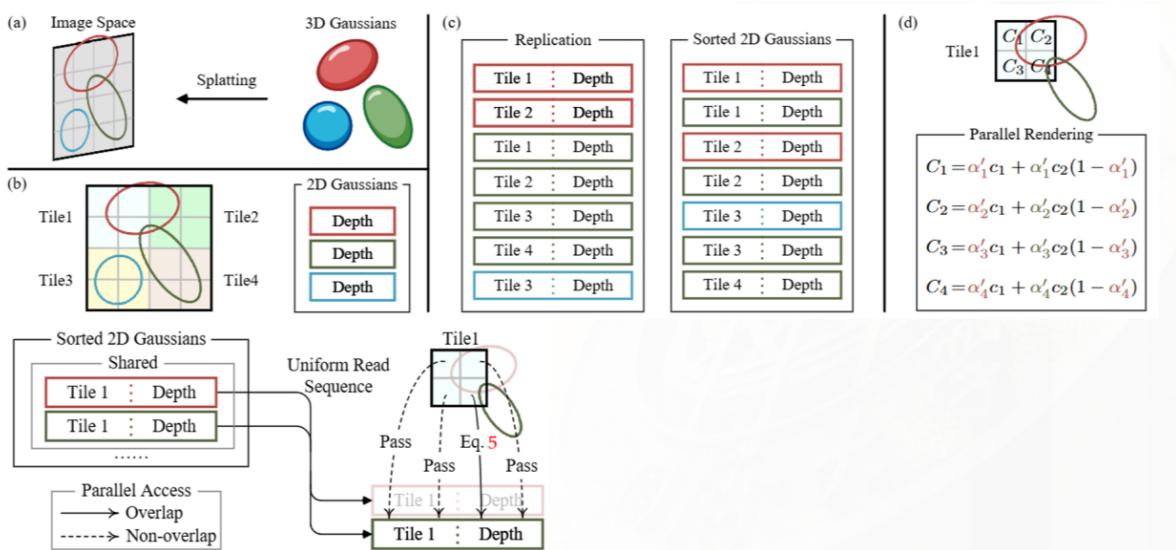


图 2. 高斯喷溅并行计算实现概述图

高斯喷溅在 Blender 数据集与 Mip-NeRF360 数据集上均表现出优异结果，其训练速度相较于其他神经辐射场提升了 50 倍，且实现了 10 秒/帧渲染速度。通过 5-10 分钟训练后，能够达到与 InstantNGP 和 Plenoxels 相当的质量，但额外的训练时间使其能够实现最先进的质量，而其他快速方法则无法做到这一点。高斯喷溅方法倾向于保留良好覆盖区域的视觉细节，即使是从远处看，这在以前的方法中并不总是能做到。



图 3. 高斯喷溅并行计算实现概述图

3 实验内容

本工作针对高斯喷溅方法进行两项实验，一是在大场景渲染模型基础上开发可视化程序用于用户交互，与其他方法对比渲染效果，二是使用预训练大模型实现可信遮罩生成用于语义渲染预测，在可交互界面下对大场景渲染结果进行实例分割预测。

3.1 实验数据与设置

实验使用 Urbanbis 数据集与 DTU 数据集。UrbanBIS 包括六个真实的城市场景，共计 25 亿个点，覆盖了 10.78 平方公里的广阔区域和 3,370 座建筑物，这些数据通过 113,346 次航拍摄影测量视图采集而成。特别地，UrbanBIS 不仅为建筑物、车辆、植被、道路和桥梁等丰富的城市对象提供了语义级别的标注，还为建筑物提供了实例级别的标注。此外，UrbanBIS 是首个引入细粒度建筑子类别的 3D 数据集，考虑了不同建筑类型的多样化形状。DTU 多视图立体 (MVS) 数据集由丹麦技术大学于 2014 年发布，包含 124 个场景，每个场景从 49 或 64 个视角拍摄，并在七种不同光照条件下获取图像，提供高分辨率图像和精确的相机参数。该数据集为多视图立体重建、计算机视觉和机器人导航等领域的研究提供了丰富的资源，广泛应用于 3D 重建、图像匹配和特征点提取等任务。渲染模型方面，本工作基于 Level of Gaussian 工作，一个基于树状索引的大场景渲染模型，本工作在此工作基础上进一步使用 opengl 技术编写可视化程序。

3.2 实验环境

操作系统: Ubuntu

实现语言: C++, CUDA

CPU: Intel(R) Xeon(R) Silver 4410Y CPU @ 3.90 GHz

GPU: NVIDIA Corporation Device 2684 (RTX 4090)

基本频率: 2.00 GHz

内核: 12

L1 缓存: 960 KB

L2 缓存: 24 MB

L3 缓存: 30 MB

内存: 263G

3.3 实验结果

3.3.1 可视化程序实现

本工作通过开发一套高度集成的可视化程序，以原始高斯数据为输入，完成了从深度估计到渲染可视化的完整处理流程。该程序的实现过程主要基于对原始高斯坐标的高效处理，结合深度估计公式与渲染技术，生成了真实感较强的多视角场景渲染结果。如图 4 所示，可视化程序能够对输入的原始高斯数据进行处理并输出多种结果，包括：

1. 渲染图像：生成逼真的场景渲染结果，展现出高质量的图像效果。
2. 可视高斯：通过直观的高斯可视化，展现原始数据的分布特征。

3. 原始深度估计：基于深度计算公式生成的初步深度估计结果。
4. 无偏深度估计：改进后的深度计算方法生成的优化深度结果，进一步提升了场景的深度预测效果。

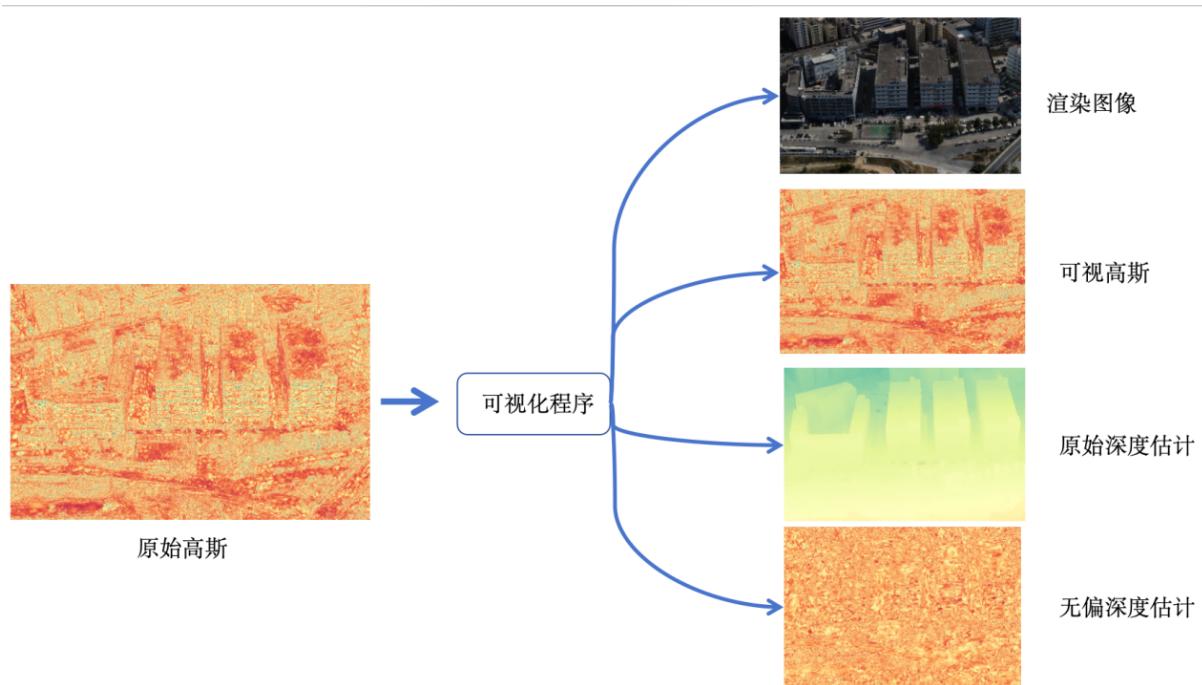


图 4. 可视化程序功能概览

在基础可视化程序的基础上，本工作进一步实现了多个功能用于交互，包括：

1. 视角拖拽与放缩：通过鼠标拖拽实现视角的自由切换，支持场景的缩放操作，并实时显示帧率。
2. 多视图切换：用户可在不同视图模式之间切换，快速获取多种表现形式下的渲染结果。
3. 深度变换渲染：利用深度估计结果动态调整渲染场景，从而呈现更为丰富的深度视觉效果。
4. 轨迹生成与视频渲染：基于用户指定的控制点生成样条插值轨迹，沿轨迹对场景进行渲染，并生成动态视频输出。

以上功能的实现不仅提升了程序的实用性，还使用户能够直观地探索复杂场景中的深度信息和渲染特性。



图 5. 可视化程序多视图显示

本项目使用 Urbanbis 数据集中的 Yingrenshi 场景对可视化程序进行了充分测试。从测试结果可以看出，程序通过高斯喷溅技术较好地完成了真实场景的渲染任务。特别是在多视角和不同缩放比例下，渲染结果展现了较高的稳定性与细节表现力。

我们在 UrbanScene3D 数据集上进行了广泛的实验评估，重点关注了 Yingrenshi 场景。我们

Method	Lihu			Yingrenshi		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
I-NGP	19.37	0.670	0.618	19.84	0.648	0.613
G-NeRF	18.81	0.662	0.670	19.64	0.683	0.677
3D-GS	16.94	0.680	0.589	18.18	0.695	0.547
我们的方法	21.75	0.799	0.318	22.84	0.796	0.331

表 1. 在 Urbanbis 数据集上的定量分析

将所提出的方法与多个最先进的方法进行了对比，包括 Instant-NGP (I-NGP)、Grid-NeRF (G-NeRF) 和 3D Gaussians (3D-GS)，使用了三个关键指标进行评估：PSNR（峰值信噪比）、SSIM（结构相似度指数）和 LPIPS（学习感知图像块相似度）。实验结果表明，我们的方法在所有评估指标上都取得了优异的表现。具体而言，在 Yingrenshi 场景中，我们的方法达到了 22.84 的 PSNR、0.796 的 SSIM 和 0.331 的 LPIPS，显著优于第二好的方法。较高的 PSNR 和 SSIM 值表明了更好的图像质量和结构保持性，而较低的 LPIPS 分数则表明了更好的感知质量。这些改进在数据集的不同场景中都保持一致，包括 Lihu 场景，我们的方法同样展现出显著优势，获得了 21.75 的 PSNR、0.799 的 SSIM 和 0.318 的 LPIPS。定量结果验证了我们基于高斯喷溅的渲染方法的有效性，特别是在维持多视角和不同缩放比例下的视觉质量方面。

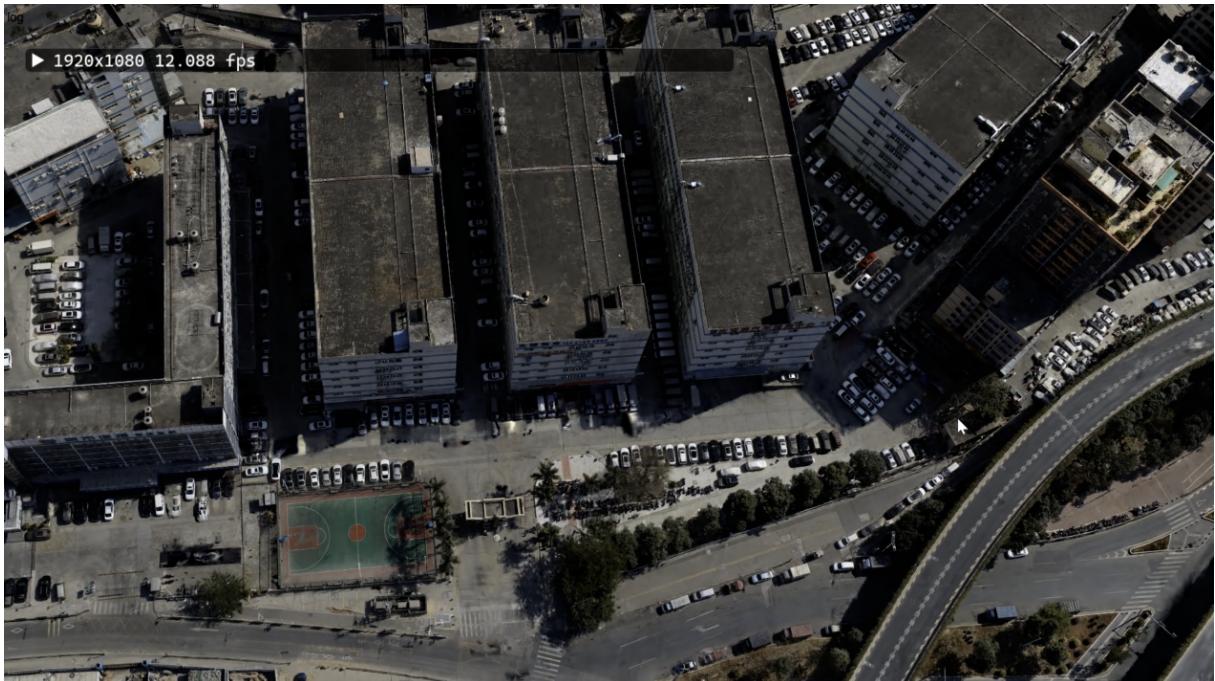


图 6. 可视化程序多视图显示

图 6 显示，程序生成的多视图渲染结果展示了不同视角下的高斯可视化和深度估计结果，清晰地反映了数据集的场景特征。这些结果证明了可视化程序在高斯数据处理、深度估计以及渲染质量方面的有效性和适用性。

通过上述可视化实现与扩展功能，本程序为深度估计与渲染领域的研究提供了重要支持，并展现了在复杂场景下的应用潜力。

3.3.2 大场景实例分割渲染探索

本工作进一步使用预训练大模型进行遮罩生成用于语义渲染预测，首先将单一轨迹渲染图像流进行提取，采用多帧联合学习思路进行局部帧的联合。

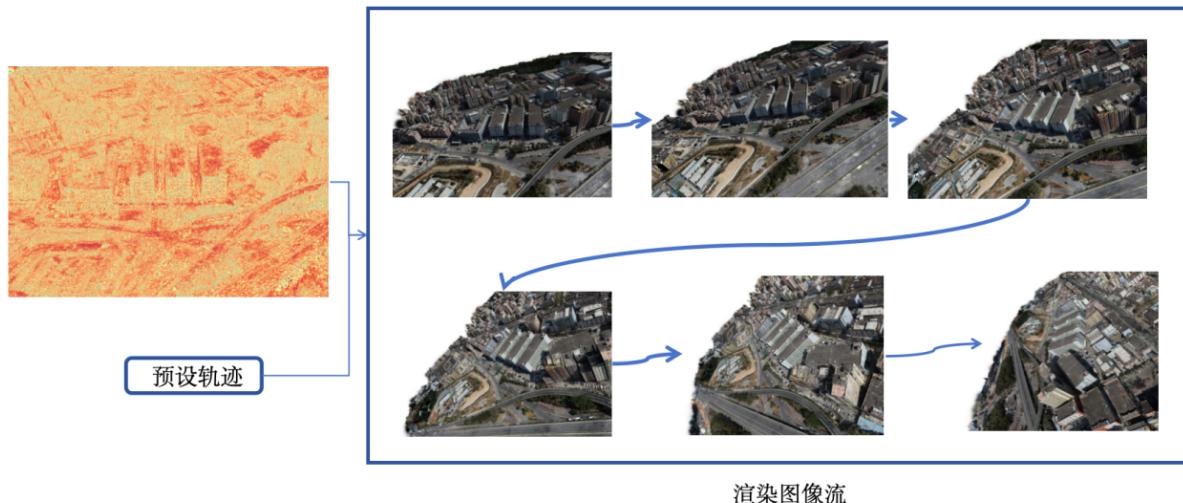


图 7. 可视化程序多视图显示

下一步，本工作使用实例分割大模型 sam2 对渲染视频流进行分割遮罩自动预测，生成原始分割结果，此时的分割遮罩在多帧序列中存在大量的不一致，在单帧中存在重叠，残缺现象，但对物体整体轮廓可以进行分辨提取。

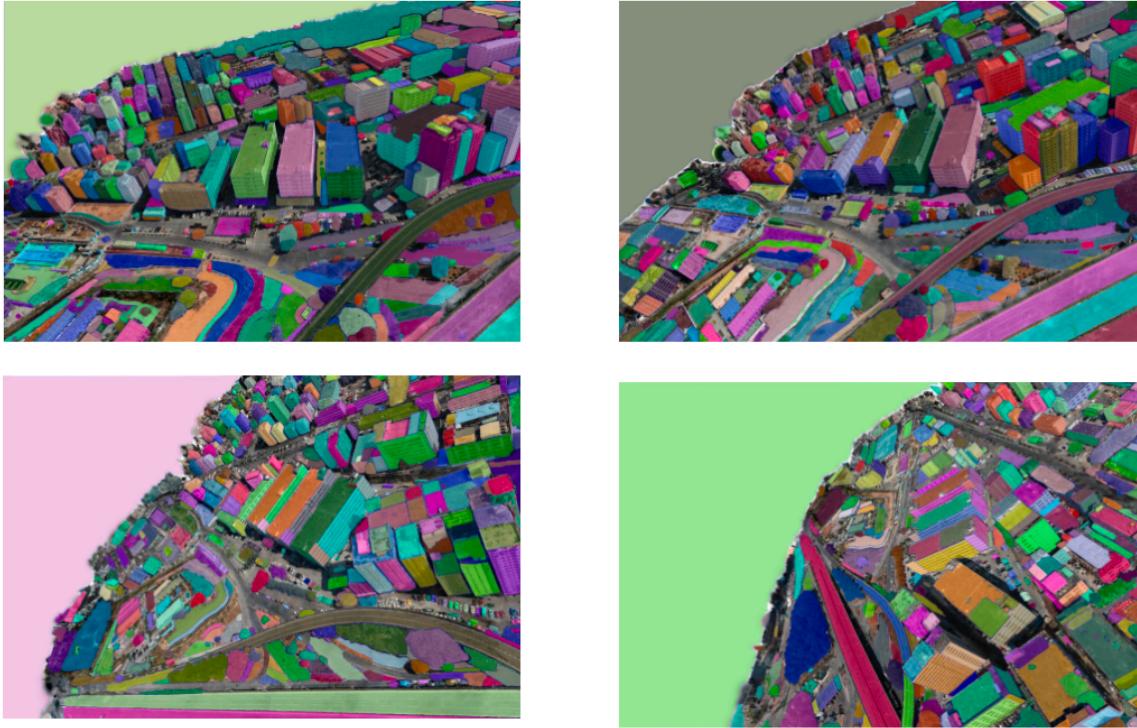


图 8. 可视化程序多视图显示

进一步，本工作使用单帧图像优化，通过多多个同一帧的预测结果进行集合操作，使用图算法进行合并操作，并使用渲染深度预测进行辅助，实现了单个物体的遮罩合并，至此，本工作实现了优秀的轨迹渲染场景的单帧可信遮罩生成，可高效应用于场景实例分割与提取工作。

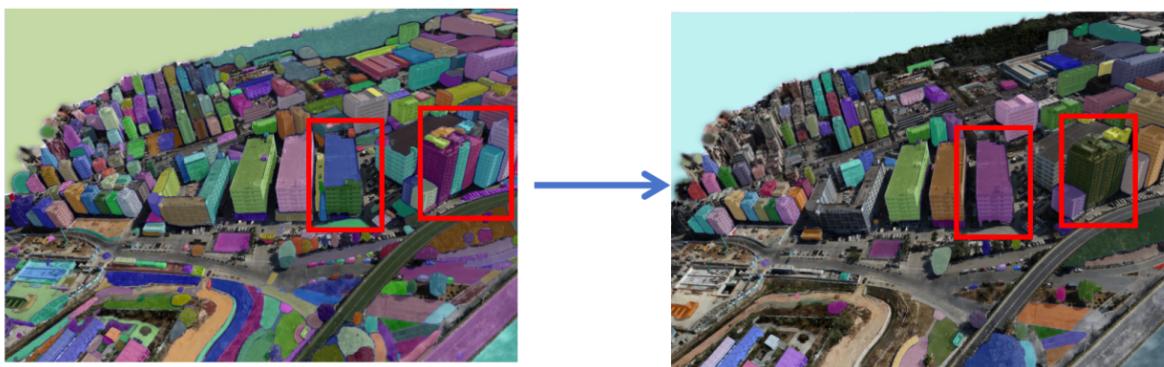


图 9. 可视化程序多视图显示

4 总结与展望

本工作在大场景实时渲染技术基础上，通过引入基础大模型，尝试在预训练大场景下进行实例分割渲染。首先，在复现大场景渲染模型的基础上，开发了一个可视化程序，能够通过高斯坐标渲染并记录中间结果，进一步实现了多功能交互，包括视角拖拽、深度变换渲染、轨迹生成等。实验结果表明，该程序能够在多视角与不同缩放比例下，生成高质量的场景渲

染效果。随后，利用预训练的大模型（如 SAM2）进行实例分割渲染，尽管初步分割结果存在不一致与重叠问题，通过多帧联合优化与深度辅助技术，最终成功生成了精确的遮罩，提升了分割精度。该方法在大场景实例分割与提取中的应用前景广泛，为复杂场景的高效处理提供了新的思路。

参考文献

- [1] Jonathan T Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P Srinivasan. Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5855–5864, 2021.
- [2] Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5470–5479, 2022.
- [3] Zhimin Fan, Jie Guo, Yiming Wang, Tianyu Xiao, Hao Zhang, Chenxi Zhou, Zhenyu Chen, Pengpei Hong, Yanwen Guo, and Ling-Qi Yan. Specular polynomials. *ACM Transactions on Graphics (TOG)*, 43(4):1–13, 2024.
- [4] Randima Fernando et al. *GPU gems: programming techniques, tips, and tricks for real-time graphics*, volume 590. Addison-Wesley Reading, 2004.
- [5] Michael Goesele, Noah Snavely, Brian Curless, Hugues Hoppe, and Steven M Seitz. Multi-view stereo for community photo collections. In *2007 IEEE 11th International Conference on Computer Vision*, pages 1–8. IEEE, 2007.
- [6] Steven J Gortler, Radek Grzeszczuk, Richard Szeliski, and Michael F Cohen. The lumi-graph. In *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*, pages 453–464. 2023.
- [7] Antoine Guédon and Vincent Lepetit. Sugar: Surface-aligned gaussian splatting for efficient 3d mesh reconstruction and high-quality mesh rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5354–5363, 2024.
- [8] Peter Hedman, Julien Philip, True Price, Jan-Michael Frahm, George Drettakis, and Gabriel Brostow. Deep blending for free-viewpoint image-based rendering. *ACM Transactions on Graphics (ToG)*, 37(6):1–15, 2018.
- [9] Peter Hedman, Tobias Ritschel, George Drettakis, and Gabriel Brostow. Scalable inside-out image-based rendering. *ACM Transactions on Graphics (TOG)*, 35(6):1–11, 2016.

- [10] Peter Hedman, Pratul P Srinivasan, Ben Mildenhall, Jonathan T Barron, and Paul Debevec. Baking neural radiance fields for real-time view synthesis. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5875–5884, 2021.
- [11] Philipp Henzler, Niloy J Mitra, and Tobias Ritschel. Escaping plato’s cave: 3d shape from adversarial rendering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9984–9993, 2019.
- [12] Marc Levoy and Pat Hanrahan. Light field rendering. In *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*, pages 441–452. 2023.
- [13] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021.
- [14] Eric Penner and Li Zhang. Soft 3d reconstruction for view synthesis. *ACM Transactions on Graphics (TOG)*, 36(6):1–11, 2017.
- [15] Xing Shen, Runyuan Cai, Mengxiao Bi, and Tangjie Lv. Preconditioned nonlinear conjugate gradient method for real-time interior-point hyperelasticity. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–11, 2024.
- [16] Noah Snavely, Steven M Seitz, and Richard Szeliski. Photo tourism: exploring photo collections in 3d. In *ACM siggraph 2006 papers*, pages 835–846. 2006.
- [17] Ayush Tewari, Justus Thies, Ben Mildenhall, Pratul Srinivasan, Edgar Tretschk, Wang Yifan, Christoph Lassner, Vincent Sitzmann, Ricardo Martin-Brualla, Stephen Lombardi, et al. Advances in neural rendering. In *Computer Graphics Forum*, volume 41, pages 703–735. Wiley Online Library, 2022.